## AMENDMENTS TO THE CLAIMS:

This listing of claims will replace all prior versions, and listings, of claims in the application:

1. (Amended) Apparatus for processing data, said apparatus comprising:

processing logic operable to perform processing operations under control of program instructions and subject to interruption by a plurality of interrupt events; and

a nested interrupt controller operable to control nested execution of a plurality of active interrupt handling programs that are started and uncompleted and which is responsive to priorities associated with respective ones of said plurality of interrupt handling programs to control pre-emption of one or more active interrupt handling programs by a pending interrupt handling program; wherein

a priority of a given active interrupt handling program is alterable whilst said given active interrupt handling program is started and uncompleted; and

said nested interrupt controller is operable:

(i) to permit ~~a~~ said pending interrupt handling program to pre-empt a plurality of active interrupt handling programs if a priority associated with said pending interrupt handling program is higher than a highest priority associated with any of said plurality of active interrupt programs; and

(ii) to prevent said pending interrupt handling program from pre-empting said plurality of active interrupt handling programs if said priority associated with said pending interrupt handling program is less than said highest priority associated with any of said plurality of active interrupt handling programs.

2. (Amended) ~~Apparatus~~ The apparatus as claimed in claim 1, wherein said nested interrupt controller is operable to prevent said pending interrupt handling program from pre-empting said plurality of active interrupt handling programs if said priority associated with said pending interrupt handling program is equal to a highest priority associated with any of said plurality of active interrupt handling programs.

3. (Amended) ~~Apparatus~~ The apparatus as claimed in claim 1, wherein a priority of an interrupt handling program is set by a programmable priority level associated with said interrupt handling program.

4. (Amended) ~~Apparatus~~ The apparatus as claimed in claim 1, wherein said pending interrupt handling program corresponds to one of:

a newly detected interrupt event; and

one or more previously detected interrupt event for which a corresponding interrupt handling program has not started execution.

5. (Amended) ~~Apparatus~~ The apparatus as claimed in claim 1, wherein said nested interrupt controller is responsive to stored values corresponding to respective ones of a plurality of interrupt events and indicative of:

a corresponding interrupt handling program being pending or active~~;~~, and

~~a corresponding interrupt handling program being active; and~~

a priority level associated with said corresponding interrupt handling program.

6. (Amended) ~~Apparatus~~ The apparatus as claimed in claim 1, comprising stack data storage operable to store state data corresponding to processing pre-empted by one or more active interrupt handling programs.

7. (Amended) ~~Apparatus~~ The apparatus as claimed in claim 6, wherein said highest priority level associated with any of said plurality of active interrupt handling programs is increased when free space within said stack data store falls below a predetermined free space level.

8. (Amended) ~~Apparatus~~ The apparatus as claimed in claim 7, wherein said highest priority level associated with any of said plurality of active interrupt handling programs is increased to a programmable priority level which is less than a maximum possible priority level.

9. (Amended) ~~Apparatus~~ The apparatus as claimed in claim 8, wherein said predetermined free space level is set to leave free space within said stack memory for use during pre-emption by interrupt handling programs having a priority higher than said programmable priority level.

10. (Amended) ~~Apparatus~~ The apparatus as claimed in claim 7, comprising a plurality of processors within a multiprocessing system, said stack data storage being a shared hardware stack used for communication between said plurality of processors.

11. (Amended) ~~Apparatus~~ The apparatus as claimed in claim 6, wherein said stack data storage is a stack memory.

12. (Previously Presented) A method of processing data, said method comprising the steps of:

performing processing operations under control of program instructions, said processing operations being subject to interruption by a plurality of interrupt events; and

controlling nested execution of a plurality of active interrupt handling programs that are started and uncompleted and in response to priorities associated with respective ones of said plurality of interrupt handling programs to control pre-emption of one or more active interrupt handling programs by a pending interrupt event; wherein

a priority of a given active interrupt handling program is alterable whilst said given active interrupt handling program is started and uncompleted; and

said step of controlling is operable:

(i) to permit a pending interrupt handling program to pre-empt a plurality of active interrupt handling programs if a priority associated with said pending interrupt handling program is higher than a highest priority associated with any of said plurality of active interrupt programs; and

(ii) to prevent said pending interrupt handling program from pre-empting said plurality of active interrupt handling programs if said priority associated with said pending interrupt handling program is less than said highest priority associated with any of said plurality of active interrupt handling programs.

13. (Amended) ~~A~~The method as claimed in claim 12, wherein said ~~nested interrupt controller is operable to prevent~~ said step of controlling is operable to prevent said pending interrupt handling program from pre-empting said plurality of active interrupt handling programs if said priority associated with said pending interrupt handling program is equal to a highest priority associated with any of said plurality of active interrupt handling programs.

14. (Amended) ~~A~~The method as claimed in claim 12, wherein a priority of an interrupt handling program is set by a programmable priority level associated with said interrupt handling program.

15. (Amended) ~~A~~The method as claimed in claim 12, wherein said pending interrupt handling program corresponds to one of:

a newly detected interrupt event; and

one or more previously detected interrupt event for which a corresponding interrupt handling program has not started execution.

16. (Amended) ~~A~~The method as claimed in claim 12, wherein said step of controlling is responsive to stored values corresponding to respective ones of a plurality of interrupt events and indicative of:

a corresponding interrupt handling program being pending~~;~~ or active, and

~~a corresponding interrupt handling program being active; and~~

a priority level associated with said corresponding interrupt handling program.

17. (Amended) ~~A~~The method as claimed in claim 12, wherein stack data storage is operable to store state data corresponding to processing pre-empted by one or more active interrupt handling programs.

18. (Amended) ~~A~~The method as claimed in claim 17, wherein said highest priority level associated with any of said plurality of active interrupt handling programs is increased when free space within said stack data store falls below a predetermined free space level.

19. (Amended) ~~A~~The method as claimed in claim 18, wherein said highest priority level associated with any of said plurality of active interrupt handling programs is increased to a programmable priority level which is less than a maximum possible priority level.

20. (Amended) ~~A~~The method as claimed in claim 19, wherein said predetermined free space level is set to leave free space within said stack memory for use during pre-emption by interrupt handling programs having a priority higher than said programmable priority level.